-Duser.timezone considered harmful, and why your Atlassian server shouldn't be on UTC

If you have a server hosting Jira and Confluence, what timezone do you set it to? UTC, or the timezone your users are in?

There is a serverfault discussion about this, and on the face of it, the question is why *wouldn't* you use UTC? Timezones are messy, ugly, human things. Why complicate your nice clean server with them?

In this post I'll outline why I think you should set your server to the timezone of the bulk of your users, rather than nice clean UTC, and specifically why, if you find yourself setting the `-Duser.timezone=...` parameter, you're probably making life harder for yourself than necessary.

Keeping the same timezone within a server

First, I think we can all agree, log files on the *same server* should all use the same time zone.

There are at least 3 log-generating things you'll be concerned with:

- The app (Jira, Confluence), and its log files (atlassian-jira.log, atlassian-confluence.log)
- The web server (Apache, nginx) and its log files (access.log, error.log)
- The operating system and its logs (/var/log/*) and persistent performance stats (e.g. /var/log/atop/*)

Events across these often need to be correlated - for instance, a spike in requests (access.log) triggers high server load (operating system), and eventually application errors (atlassian-jira.log). For your debugging sanity, these log files must have the same timezone.

The web server and OS logs are all in the OS timezone, obviously. How about app server logs, atlassian-jira.log and atlassian-confluence.log?

App server log timestamps will also be in the OS timezone, unless you've told Java otherwise by explicitly setting `-Duser.timezone` . To spell it out: if you've set `-Duser.timezone` , you can no longer correlate datestamps between atlassian-jira.log, access.log and the rest of the system.

So don't do that. Don't ever set `-Duser.timezone`. Leave Java in the same timezone as your OS.

Why not UTC?

There are two arguments against setting your server timezone to UTC.

The argument from ease-of-debugging

Users report problems relative to their timezone. Screenshots might be saved with timestamped filenames. "Things got really slow around 4pm" they'll say, and your job is to find what was happening on the server at 4pm.

~~I work on Linux, and my tooling hasn't evolved much since the 1970s. I don't know an easy way to convert log timestamps from UTC to something else, on the fly. If the log's timezone is that of the user, life is just easier.~~

> **2023 Update**
>
> The lnav log file utility now reinterprets timestamps using the system timezone and `TZ` environment variable (as of ticket #703,15 Aug 2023). This means if you're on a UTC server but want to see timestamps in your native timezone, just `export TZ=Australia/Sydney` or whatever before running `lnav` .

The argument from Confluence developer laziness

"But", our administrator says, "I want UTC on the server for *<insert sane, excellent reasons>*, and I use *<insert names of tool that let me view logs in any timezone, which I shall share with you in the comments>*. Why can't the OS and Java be set to UTC, and the apps be told only to display dates in a specific format?"

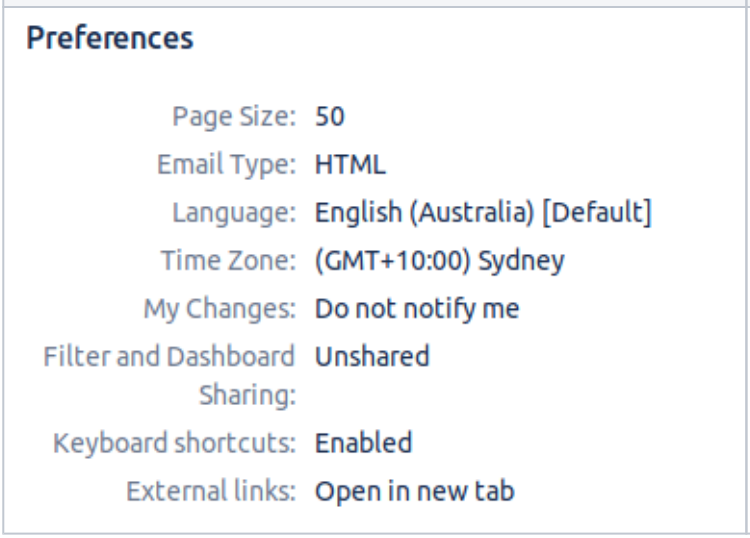
The answer is that this would be possible, except that Confluence developers failed to implement a default user timezone setting (~~CONFSERVER-16057~~ - Allow Confluence timezone to be set as a global preference CLOSED).

To review, the timezone users see in Jira and Confluence comes from one of various levels:

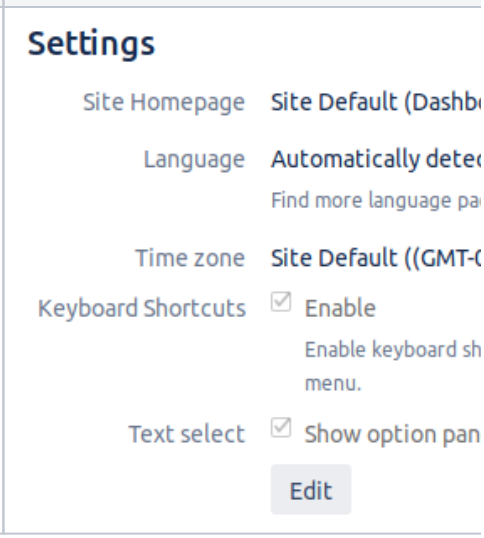
1) User preference

Users can explicitly set their timezone in their profile:

Jira



Confluence

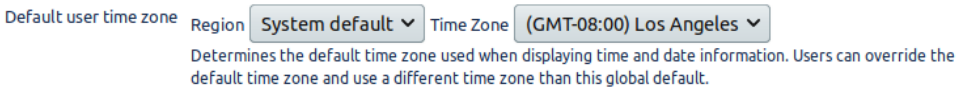


2) Default user preference

If your user preference isn't set, the **default user preference** will be in effect.

Jira

Jira conveniently lets you edit the default user timezone in General Configuration:



Confluence

Confluence does not let you edit the default timezone (~~CONFSERVER-16057~~ - Allow Confluence timezone to be set as a global pre
to editing XML files inside jars.

3) Java `user.timezone`

The default user timezone preference inherits its value from the **Java `user.timezone` Java property**.
You can set this `user.timezone` Java flag in `bin/setenv.sh`, e.g. to `-Duser.timezone=America/Los_Angeles` .

4) OS timezone

Finally, Java sets its `user.timezone` property default to the operating system's timezone:

```
$ cat /etc/timezone
Australia/Sydney
```

So in Jira, everything is great. You can have `atlassian-jira.log` use a timezone that's good for debugging (UTC if that's your preference), while the default timezone used to render dates can be something appropriate for the majority of users.

In Confluence, sadly there is only one default timezone (Java's `user.timezone`). It will have to be set to the timezone that makes sense for most users. This means `atlassian-confluence.log` will also use that timezone, and for consistency, so should the rest of the OS.

Conclusion

The conclusions are:

- Don't ever use `-Duser.timezone` as it leads to inconsistent timezones across local log files.
- Vote for ~~CONFSERVER-16057~~ - Allow Confluence timezone to be set as a global preference CLOSED (bitter laughter).
- If you running Confluence, your OS timezone should be set to what you want your default user timezone for users to be.
- If you running only Jira, I still suggest setting your OS timezone to what you want the default user timezone to be, just to make debugging simpler. However you have the option to use UTC as your OS and Java timezone, as long as you set Jira's default user timezone explicitly.